Fundamental Algorithms, Problem Set 3 Solutions

1. Write each of the following functions as $\Theta(g(n))$ where $g(n)$ is one of the standard forms: $2n^3 - 11n + 98$; $6n + 43n \lg n$; $63n^2 + 14n \lg^5 n$; $3 + \frac{5}{n}$
   `Solution:`In order, $\Theta(n^4), \Theta(n \lg n), \Theta(n^2), \Theta(1)$.

2. Illustrate the operation of `RADIX-SORT` on the list: COW, DOG, SEA, RUG, ROW, MOB, BOX, TAB, BAR, EAR, TAR, DIG, BIG, TEA, NOW, FOX following the Figure in the Radix-Sort section. (Use alphabetical order and sort one letter at a time.)
   `Solution:`From left to right:

COW	SEA	TAB	BAR
DOG	TEA	BAR	BIG
SEA	MOB	EAR	BOX
RUG	TAB	TAR	COW
ROW	DOG	SEA	DIG
MOB	RUG	TEA	DOG
BOX	DIG	DIG	EAR
TAB	BIG	BIG	FOX
BAR	BAR	MOB	MOB
EAR	EAR	DOG	NOW
TAR	TAR	COW	ROW
DIG	COW	ROW	RUG
BIG	ROW	NOW	SEA
TEA	NOW	BOX	TAB
NOW	BOX	FOX	TAR
FOX	FOX	RUG	TEA

3. Given $A[1 \cdots N]$ with $0 \leq A[I] < N^N$ for all I.

   (a) How long will `COUNTING-SORT` take?
   `Solution:`$\Theta(N^N)$ since you have to go through array C of length N^N. If you wrote $O(N^N + N)$ it is not technically wrong but it misses the point. In asymptotics we ignore the lower order terms to put things in the right form. Thus we want to write $O(N^N)$, in this case $\Theta(N^N)$ since you do indeed need to go through array C.

(b) How long will `RADIX-SORT` take using base N?
`Solution:`Now C has length N do for each digit this is a linear sort, $\Theta(N)$. However, there are N digits so that the total time is $\Theta(N^2)$.

4. `Just for Fun:`[1] Your instructor shares his family name (no relation, unfortunately!) with one of the most famous women of the twentieth century. Who was that woman?
Diana, Princess of Wales. Married and Divorced from Prince Charles, mother of William and Harry. Death in tragic car crash in Paris. Maiden name: Diana Spencer

5. Prof. Squander decides to do Bucket Sort on n items with n^2 buckets while his student Ima Hogg decides to do Bucket Sort on n items with $n^{1/2}$ buckets. Assume that the items are indeed uniformly distributed. Assume that Ima's algorithm for sorting inside a bucket takes time $O(m^2)$ when the bucket has m items.

(a) Argue that Prof. Squander has made a poor choice of the number of buckets by looking analyzing the time of Bucket Sort in his case.
`Solution:`The time will be $O(n^2)$ since one has to pass through the buckets to link them up. Note that even though most of the buckets are empty you don't know which one's are empty so you have to check each one.

(b) Argue that Ima has made a poor choice of the number of buckets by looking analyzing the time of Bucket Sort in her case.
`Solution:`Lets say each of the $n^{1/2}$ buckets had around $n^{1/2}$ items. (Indeed, with high probability that will be the case.) Then sorting each bucket (under our assumption) takes $O((n^{1/2})^2) = O(n)$ so the total time for bucket sorting would be $O(n^{3/2})$.

(c) (Thanks, Dianjing!) Compare the *space* usage of the above two solutions and of using n buckets. Which solutions (if any) use substantially more or less space than which other solutions?
`Solution:`Squander has n^2 space since there is an array of size n^2 – even though most elements of the array are empty lists they still each take constant space. Thats bad. Ima is *not* substantially better than n buckets since the total length of her linked lists is n. So Ima and n buckets both have similar $\Theta(n)$ space whereas

[1] Just for Fun problems are not graded!

Squander has squandered his space, using $\Theta(n^2)$. (True, using n buckets squanders $\sim 0.36n$ space on empty buckets but that is not a substantial amount and we ignore it in Θ-land.)

6. Analyze the time for the following algorithm. Your answer should be $\Theta(g(n))$ for some nice function $g(n)$ (* is product, ++ is increment 1)
FOR $I = 1$ TO N
$J = 1$
WHILE $J * J < I$
$J++$
END WHILE
END FOR
`Solution:`The WHILE loop takes time $\Theta(\sqrt{I})$ as it ends when J reaches $\sqrt{I}$. So the total time is $\sqrt{1} + \ldots + \sqrt{N}$. We have the upper bound
$$\sqrt{1} + \ldots + \sqrt{N} \leq N\sqrt{N}$$
and, by halving, the lower bound
$$\sqrt{1} + \ldots + \sqrt{N} \geq (N/2)\sqrt{N/2} = \frac{1}{2\sqrt{2}} N\sqrt{N}$$
The time has been sandwiched and the total time is $\Theta(N^{3/2})$.